

ÉCOLE POLYTECHNIQUE

INF552

ANALYSE D'IMAGES ET VISION PAR ORDINATEUR

Graphcut et photomontage

Binôme :

Diego DE SOUZA
Henrique GASPARINI FIUZA DO
NASCIMENTO

Professeur :
Renaud KERIVEN



6 janvier 2017

1 Introduction

Dans ce projet, on propose une interface graphique permettant de recoller des images en résolvant un problème de flot maximum, souvent appelé un problème de graphcut.

Notre interface offre à l'utilisateur deux options : il peut soit recoller deux images différentes soit choisir une seule image et la répliquer plusieurs fois. Les résultats obtenus ont été satisfaisants et sont dans le corps de ce rapport.

2 Algorithme

Le cœur de l'algorithme consiste en approcher le problème de recollage de deux images comme un problème de graphes et est inspiré surtout de l'article de la première référence.

On construit un graphe dans lequel chaque pixel de l'image correspond à un sommet et chaque paire de pixels voisins est reliée par une arête d'un poids à être spécifié : un poids plus lourd doit correspondre à une transition plus importante d'un sommet à l'autre (dans le cas où chaque sommet serait pris d'une image source différente). On définit deux sommets extérieurs qui seront un sommet source et un sommet destination et on les relie à certains sommets par des arêtes de poids distincts : une arête de poids important reliant un pixel à la source révèle une tendance à prendre le pixel de la première image, alors que une arête reliant un pixel à la destination révèle une tendance à prendre le pixel de la deuxième image.

Ainsi, si l'on utilise la modélisation décrite ci-dessus pour recoller deux images, la région de transition d'une image vers l'autre correspond visuellement à une coupe, ce qui est vrai aussi pour le graphe correspondant, lequel est divisé dans une partie qui est reliée au sommet source et une deuxième partie qui est reliée au sommet destination.

On dit qu'une transition est plus douce qu'une autre transition si sa somme des poids des arêtes coupées est moins importante. D'après cette interprétation, le recollage le plus doux correspond à la coupe minimale.

Le choix des poids des arêtes peut se faire de différentes manières, d'après ce qu'on a vu dans les articles étudiés.

Dans notre approche, nous permettons à l'utilisateur de choisir l'une parmi plusieurs combinaisons linéaires de deux formules différentes pour les distances entre pixels voisins s et t provenant de deux images A et B :

$$d1(s, t, A, B) = \|A(s) - B(s)\| + \|A(t) - B(t)\| \quad (1)$$

$$d2(s, t, A, B) = \|G_A(s) - G_B(s)\| + \|G_A(t) - G_B(t)\| \quad (2)$$

où G_A et G_B sont les matrices de gradients des images A et B respectivement. $A(s)$ et $B(s)$ sont les vecteurs RGB du pixel s des images A et B respectivement.

Le choix de la combinaison correspond au choix du paramètre lambda :

$$((\lambda_{max} - \lambda)d1(s, t) + (\lambda)d2(s, t)) / \lambda_{max} \quad (3)$$

Dans cette formule, λ_{max} a été choisi arbitrairement comme égal à 10.

Le choix des poids pour les arêtes reliant les pixels à la source et à la destination a été plus simple. Pour les points qui ne sont pas dans la région d'overlapping on est sûre de l'image qu'il doit appartenir. Ainsi, on attribue des poids infinis aux arêtes correspondants.

Le paramètre Type défini si la coupe sera horizontale ou bien vertical.

Le code rectangleOverlap.cpp est le responsable de déterminer la région d'overlapping entre deux images. Ces images peuvent être déplacées ou non par rapport à l'origine.

3 Logiciel

Pour lancer le programme qui a été fait pour ce projet, on doit passer comme arguments de la fonction main soit deux chemins, chacun correspondent à des images différentes pour le mode de recollage de deux images, soit un seul chemin correspondant à l'image qui sera recollée plusieurs fois.

Après le démarrage du logiciel l'interface graphique 1 est lancée et l'utilisateur peut choisir les paramètres utilisés par l'algorithme. Le photomontage

est recalculé après chaque modification des paramètres.



FIGURE 1 – interface graphique

4 Résultats

Comme décrit dans la section 3, il existe deux modes d’opération : soit recoller deux images différentes soit recoller la même image plusieurs fois. Les photomontages 4, 7 et 8 sont des résultats du premier mode d’opération mentionné. L’image à droite de ces photomontages montre la coupe produite par l’algorithme de graphcut. Pour chaque photomontage on montre aussi les images qui ont été utilisées comme entrée du problème.



FIGURE 2 – Raft



FIGURE 3 – River

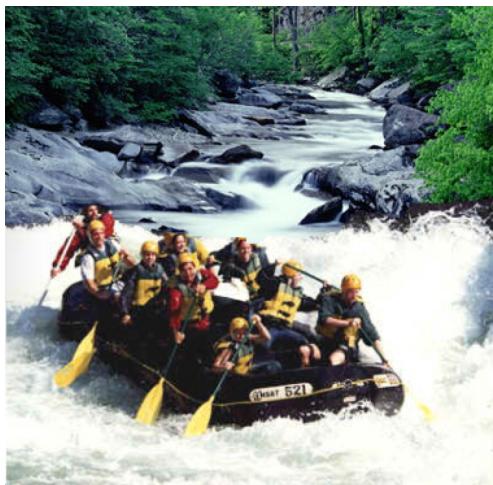


FIGURE 4 – Photomontage des images 2 et 3

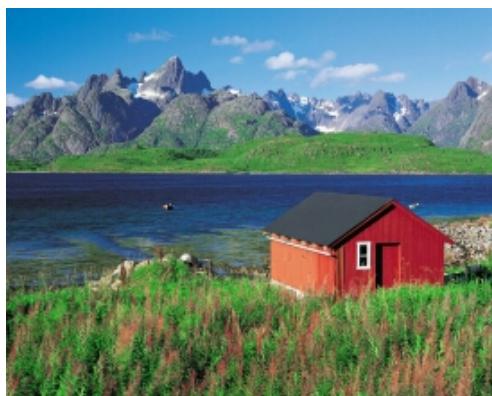


FIGURE 5 – Hut

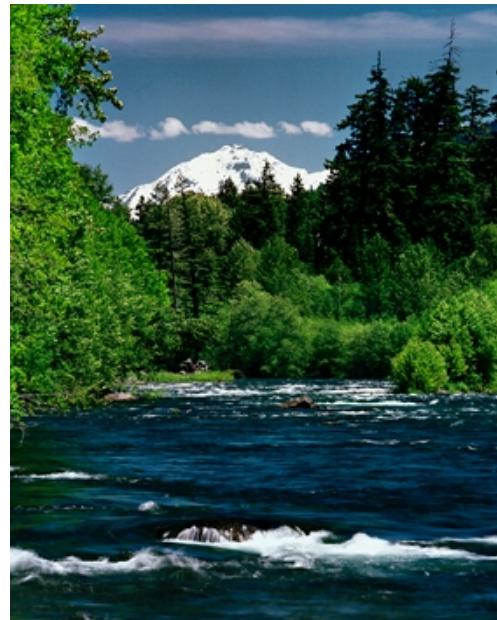


FIGURE 6 – Mountain

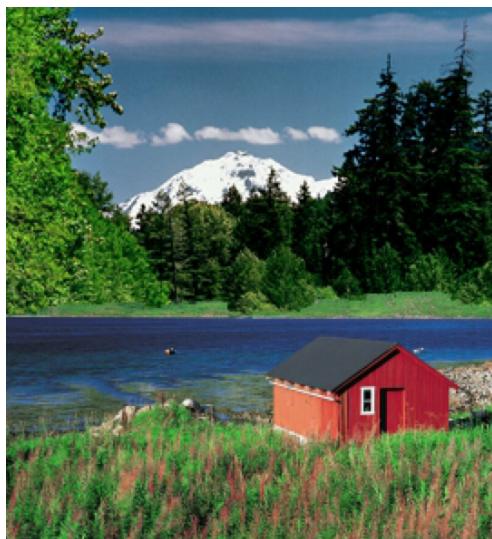


FIGURE 7 – Photomontage des images 5 et 6



FIGURE 8 – Photomontage des images 5 et 6

Les autres photomontages utilisent le même algorithme de graphcut, mais au lieu d'avoir une seule image, la même image est recollée plusieurs fois. De meilleurs résultats auraient pu être obtenus en faisant de rotation ou en appliquant de techniques de perspective avant chaque recollage de la même image.

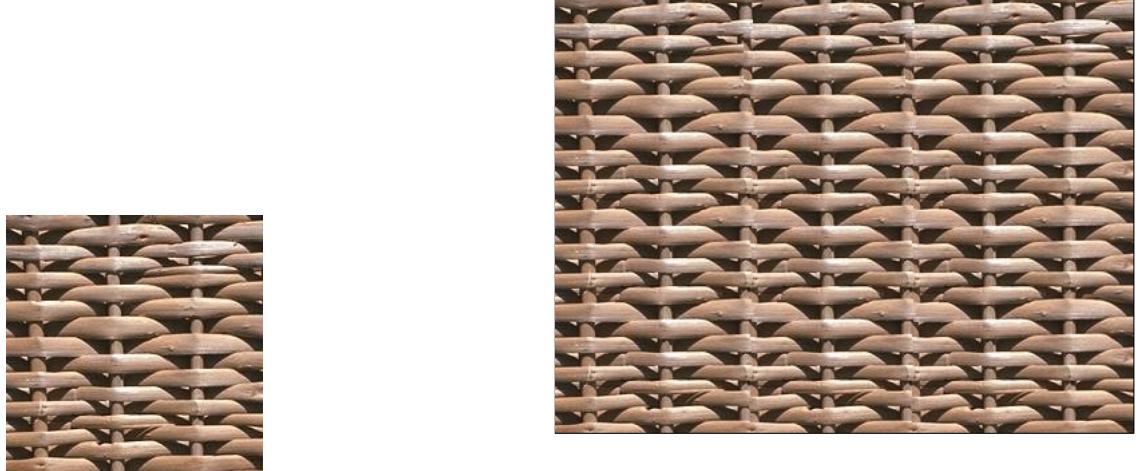


FIGURE 9 – Photomontage de la même image



FIGURE 10 – Photomontage de la même image

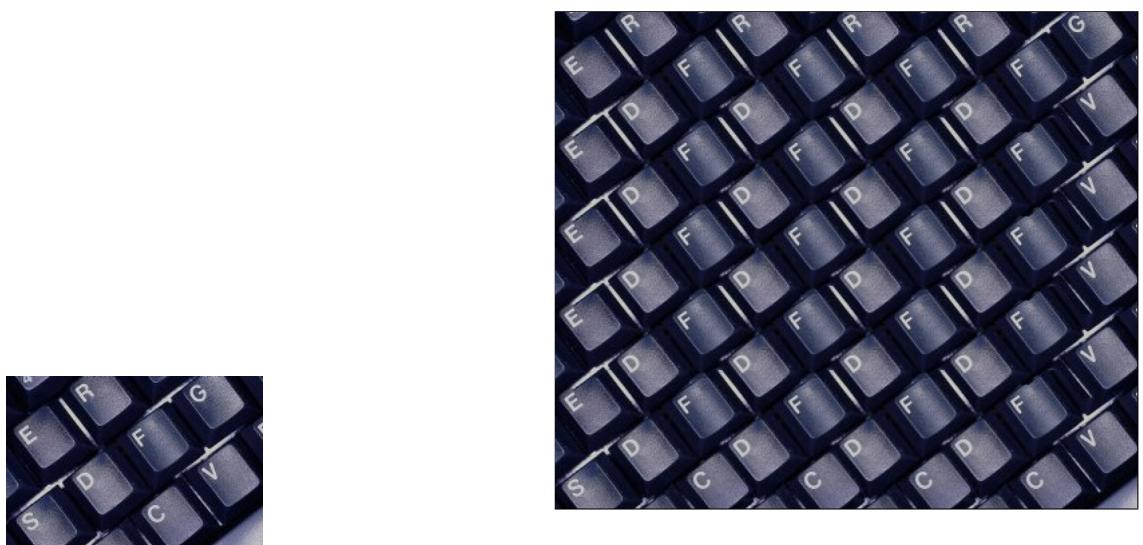


FIGURE 11 – Photomontage de la même image

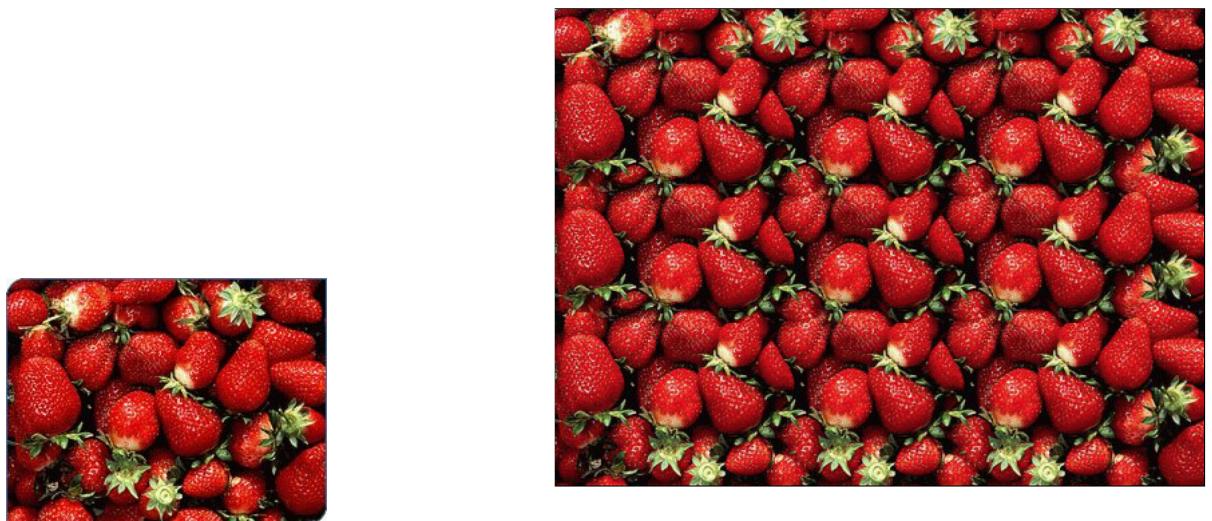


FIGURE 12 – Photomontage de la même image



FIGURE 13 – Photomontage de la même image

5 Bibliographie

- A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. *ACM Trans. Graph.*, 23(3) :294–302, Aug. 2004. ISSN 0730-0301. doi : 10.1145/1015706.1015718. URL <http://doi.acm.org/10.1145/1015706.1015718>.
- V. Kwatra, A. Schödl, I. Essa, G. Turk, and A. Bobick. Graphcut textures : Image and video synthesis using graph cuts. *ACM Trans. Graph.*, 22(3) : 277–286, July 2003. ISSN 0730-0301. doi : 10.1145/882262.882264. URL <http://doi.acm.org/10.1145/882262.882264>.